

PATENT ABSTRACTS OF JAPAN

(11)Publication number : 03-228434

(43)Date of publication of application : 09.10.1991

(51)Int.Cl.

H03M 7/50

(21)Application number : 02-024073

(71)Applicant : NEC CORP

(22)Date of filing : 01.02.1990

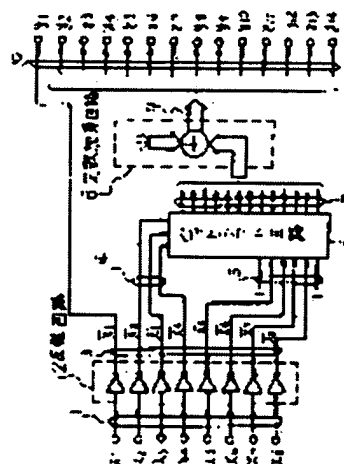
(72)Inventor : ICHIHARA MASAKI

(54) CODE CONVERTING CIRCUIT

(57)Abstract:

PURPOSE: To execute the code conversion from a μ LawPCM code of 8 bits to a linear code and to miniaturize the chip size without using a ROM by using an inversion circuit, a bit shift circuit and an adder.

CONSTITUTION: This code converting circuit is provided with an inversion circuit 2 of 8 bits, a bit shift circuit 6 for inputting the lower 4 bits of the inversion circuit 2, and also, executing a bit shift operation in the upper digit direction extending from '0' bit to 7 bits by three control signals formed by three bits in the upper 4 bits, and a constant adding circuit 8 for adding a prearranged constant to the shift circuit 6. In such a way, it is realized to miniaturize a chip size and to reduce the cost of the chip without using a ROM, and the code conversion of a μ LawPCM code of 8 bits and a linear code is executed.



⑨ 日本国特許庁(JP)

⑩ 特許出願公開

⑫ 公開特許公報(A) 平3-228434

⑤ Int. Cl.⁵

識別記号

庁内整理番号

⑬ 公開 平成3年(1991)10月9日

H 03 M 7/50

6832-5J

審査請求 未請求 請求項の数 5 (全6頁)

⑭ 発明の名称 符号変換回路

⑯ 特 願 平2-24073

⑰ 出 願 平2(1990)2月1日

⑱ 発 明 者 市 原 正 貴 東京都港区芝5丁目33番1号 日本電気株式会社内

⑲ 出 願 人 日本電気株式会社 東京都港区芝5丁目7番1号

⑳ 代 理 人 弁理士 内 原 晋

明 細 書

1. 発明の名称

符号変換回路

2. 特許請求の範囲

1. 8ビットの反転回路と、前記反転回路の下位4ビットを入力し且つ上位4ビットのうち3ビットで形成した3つの制御信号によって0ビットから7ビットまでの上位桁方向のビットシフト演算を行うビットシフト回路と、前記ビットシフト回路の出力に既定の定数の加算を行う定数加算回路とを含むことを特徴とする符号変換回路。

2. μ Law PCM符号で構成された8ビットの入力信号を反転する反転回路と、前記反転回路の出力信号の下位4ビットに論理「1」レベルの信号を上下1ビットづつ付加して生成した6ビットの信号を入力し前記反転回路出力信号の上位から2乃至4ビット目の値に応じて上位桁方

向に0ビットから7ビットまでのビットシフトを行った結果13ビットの信号として出力するビットシフト回路と、前記ビットシフト回路の13ビットの出力信号に既定の定数値を加算する定数加算器とを有し、前記定数加算器の13ビットの出力信号の上位桁に前記反転回路の出力信号の最上位ビットをサインビットとして付加することにより合わせて14ビットの擬形符号を出力することを特徴とする符号変換回路。

3. 請求項2記載のビットシフト回路は、1ビットシフト部と2ビットシフト部および4ビットシフト部を順次接続して構成したことを特徴とする符号変換回路。

4. 請求項2記載の定数加算回路は、最下位ビットを除く各入力ビットに1もしくは0を加算する第一の全加算器と、最下位ビットにサインビットの反転信号を加算する第二の全加算器とを有して構成したことを特徴とする符号変換回路。

5. μ Law PCM符号で構成された8ビットの入力信号を反転する反転回路と、前記反転回路の

出力信号の下位4ビットに論理「1」レベルの信号を上下1ビットづつ付加して生成した6ビットの信号を入力し前記反転回路出力信号の上位から2乃至4ビット目の値に応じて上位桁方向に0ビットから7ビットまでのビットシフトを行った結果13ビットの信号として出力するビットシフト回路と、前記ビットシフト回路の13ビットの出力信号に既定の定数値を加算する定数加算器と、前記定数加算器の13ビットの出力信号の各々および前記反転回路の最上位ビットの排他的論理和をとる1の補数回路とを有し、前記1の補数回路の13ビットの出力信号の上位桁に前記反転回路の出力信号の最上位ビットをサインビットとして付加することにより合わせて14ビットの線形符号を出力することを特徴とする符号変換回路。

3. 発明の詳細な説明

〔産業上の利用分野〕

本発明は符号変換回路に関し、特に8ビットの

図である。

第5図に示すように、従来の符号変換回路はROMを使用している。すなわち、8ビットの μ Law符号($x_1 \sim x_8$)24をROM22のアドレス21として入力し、そのアドレスに対応する記憶データをデータ出力23に14ビットの線形符号($y_1 \sim y_{14}$)25として読み出すものである。この符号変換回路は8ビットに対応する256個のデータをそのまま市販のPROMやマスクROMに書き込むだけでよく、ディスクリートな部品を用いた装置の設計においては、非常に簡単で且つ安価な構成と言える。

〔発明が解決しようとする課題〕

しかしながら、近年の半導体技術の進歩により、デジタル信号処理の分野においても、装置レベルでのLSI化が進展している。しかも、この種の符号変換回路はディスクリートな部品ではなく、LSIチップ内の単なる一機能として設計する必要が生じてきている。これを従来技術のようにROMを用いて構成すると、符号の個性を利用して

μ LawPCM符号を14ビットの線形符号に変換するための符号変換回路に関する。

〔従来の技術〕

従来、かかる8ビットの μ LawPCM符号(以下、単に μ Law符号と称す)および14ビットの線形符号(以下、単に線形符号と称す)の対応は、CCITT勧告-Q711において定められている。

この μ Law符号の汎用8ビットPCMコーデックは、安価で且つ手がかるに利用出来るため、デジタル信号処理の分野等におけるアナログ信号とデジタル信号とのインターフェイス用に盛んに使用されている。この μ Law符号はデータ量を圧縮するために非線形な符号化方式を採用しているため、そのままの形でデジタル信号処理に必要な加減乗除演算を行うことはできない。そこで、デジタル信号処理を可能にするため、 μ Law符号を線形符号に変換する符号変換回路が使用されている。

第5図はかかる従来の一例を示す符号変換回路

データ量を圧縮しても128ワード×14ビットの記憶容量が必要であり、LSIチップでは大面積を占有してしまふ。従って、従来の符号変換回路はチップサイズを増大させ、その結果チップの単価が高くなってしまふという欠点がある。

本発明の目的は、かかるROMを用いることなくチップサイズの小型化およびチップの低価格化を実現し、 μ Law符号と線形符号の符号変換を行なうことのできる符号変換回路を提供することにある。

〔課題を解決するための手段〕

本発明の符号変換回路は、8ビットの反転回路と、前記反転回路の下位4ビットを入力し且つ上位4ビットのうち3ビットで形成した3つの制御信号によって0ビットから7ビットまでの上位桁方向のビットシフト演算を行うビットシフト回路と、前記ビットシフト回路の出力に既定の定数の加算を行う定数加算回路とを含んで構成される。

〔発明の原理〕

次に、本発明の原理について説明する。

CCITT勧告-G711.によれば、前述したように、 μ Law符号と線形符号の対応が定められている。この勧告を分析すると、 μ Law符号と線形符号の変換則は、以下のように記述できる。

まず、 μ Law符号を $X = \{x_1, x_2, x_3, x_4, x_5, x_6, x_7, x_8\}$ とする。ただし、 x_i は各ビットの値であり、 x_1 がMSB、 x_8 がLSBである。また、線形符号を $Y = \{y_1, y_2, y_3, y_4, y_5, y_6, y_7, y_8, y_9, y_{10}, y_{11}, y_{12}, y_{13}, y_{14}\}$ とする。これも μ Law符号と同様に、 y_i は各ビットの値であり、 y_1 がMSB、 y_{14} がLSBである。さらに、 μ Law符号の反転を $\bar{X} = \{\bar{x}_1, \bar{x}_2, \bar{x}_3, \bar{x}_4, \bar{x}_5, \bar{x}_6, \bar{x}_7, \bar{x}_8\}$ で表す。

これにより、 Y と \bar{X} の関係は、次式で表される。

$$y_1 = \bar{x}_1 \quad \dots \dots (1)$$

$$\sum_{i=2}^{14} y_i \cdot 2^{14-i} = 2^{(4+\bar{x}_2+2\bar{x}_3+\bar{x}_4)} \cdot (33 + \sum_{i=5}^8 \bar{x}_i \cdot 2^{8-i}) - 33 \quad \dots \dots (2)$$

第1図に示すように、本実施例は μ LawPCM符号で構成された8ビットの入力信号($x_1 \sim x_8$)を反転する反転回路2と、この反転回路1の出力信号($\bar{x}_1 \sim \bar{x}_8$)3の下位4ビット $\bar{x}_5 \sim \bar{x}_8$ に論理「1」レベルの信号を上下1ビットづつ付加して生成した6ビットの信号5を入力し且つこの反転回路2の出力信号のうち上位から2乃至4ビット $\bar{x}_2 \sim \bar{x}_4$ 目の値に応じて上位桁方向に0ビットから7ビットまでのビットシフトを行った結果13ビットの信号7として出力するビットシフト回路6と、このビットシフト回路6の13ビットの出力信号7に既定の定数値(ここでは、-33)を加算する定数加算器8とを有し、定数加算器8の13ビットの出力信号9の上位桁にさらに反転回路2の出力信号の最上位ビット \bar{x}_1 をサインビットとして付加することにより合わせて14ビットの線形符号、すなわちサイン+絶対値表示出力($y_1 \sim y_{14}$)10を出力するものである。

かかる符号変換回路において、まず入力された μ Law符号1は、反転回路2で反転され、その

次に、この式(1)の意味は、線形符号 Y の最上位ビット y_i が μ Law符号の反転 \bar{X} の最上位ビット \bar{x}_i に等しいことを表わす。また式(2)は線形符号 Y の下位13ビットの算出を表わし、その算出手順は①→②→③の順に行なわれる。

その①は \bar{X} の下位4ビットを1ビット上位桁方向にシフトした後、10進数の33に相当する2進数を加えることである。その②はこの2進数を加えた結果を、 $(\bar{x}_2, \bar{x}_3, \bar{x}_4)$ なる2進数で示される桁数だけ、上位桁方向にシフトすることである。次に、③はこの結果から10進数の33に相当する2進数を引くステップである。

本発明は、上述した式(1)と(2)のアルゴリズムを純粋な組合せ論理回路を用いて実現するものである。

[実施例]

次に、本発明の実施例について図面を用いて説明する。

第1図は本発明の第一の実施例を示す符号変換回路のブロック図である。

反転結果の内、上位1ビットはそのまま線形符号の最上位ビットとして出力される。また、ビットシフト回路6には、下位4ビットを1ビット上位桁方向にシフトし、「-33」を加えた合計6ビットの信号5を入力する。このビットシフト回路6は、 $\bar{x}_2, \bar{x}_3, \bar{x}_4$ の3本の制御信号4の値に応じて、入力信号5を第1表(ビットシフト回路6のシフト量)に示すビット数だけ、上位桁方向にシフトし、13ビットの信号として出力する。更に、定数加算回路8は、ビットシフト回路6の出力信号7に「-33」を加え、その出力13ビットを線形符号10の下位13ビットとして出力する。以上の例は、線形符号がサインビット+絶対値表示方式の場合である。

第 1 表

\bar{x}_2	\bar{x}_3	\bar{x}_4	シフト量
0	0	0	0
0	0	1	1
0	1	0	2
0	1	1	3
1	0	0	4
1	0	1	5
1	1	0	6
1	1	1	7

第2図に第1図に示すビットシフト回路図である。

第2図に示すように、このビットシフト回路6は、1ビットシフト部11と、2ビットシフト部12と4ビットシフト部13とを順次接続した3段階構成になっており、それぞれ反転回路2の出力3のうちの $\overline{x4}$ 、 $\overline{x3}$ 、 $\overline{x2}$ 信号が「1」レベルの時にシフト動作を行うようになっている。このビットシフト回路6のそれぞれのユニット11~13は、トランスファゲートやアンドオアゲートなどの素子により構成できる。

また、第3図は第1図に示す定数加算回路図である。

第3図に示すように、この定数加算回路8は13個の全加算器(FULL ADDER)14を用いて、ビットシフト回路6からの出力信号7に「33」または、「34」の2の補数値を加え、13ビットの信号9として出力している。この定数加算器8における「-33」と「-34」の切り替えは、加算する定数の最下位ビットを「1」にするか「0」

回路2の出力信号3の上位から2乃至4ビット($\overline{x2} \sim \overline{x4}$)4目の値に応じて上位桁方向に0ビットから7ビットまでのビットシフトを行った結果を13ビットの信号7として出力するビットシフト回路6と、このビットシフト回路6の13ビットの出力信号7に既定の定数値を加算する定数加算器8と、この定数加算器8の13ビットの出力信号9の各々および反転回路2の最上値ビット $\overline{x1}$ の排他的論理和をとる1の補数回路17とを有し、この1の補数回路17の13ビットの出力信号 $y2 \sim y14$ の最上位ビット y 、サインビット16として付加することにより、合わせて14ビットの綴形符号を出力するものである。

この符号変換回路において反転を行う回路として、排他的論理和による1の補数回路17を用いた点が前述した第一の実施例と比較して異なっている。また、反転後に「1」を加えるためには、別途加算器が必要であるが、回路規模を抑えるため、等価な手段として定数加算回路8で加算する定数を「-33」と「-34」の2通り用意しておき、

にするかで行っている。この値9は、サインビット16をインバータ15で反転した信号そのもので与えられる。尚、第1図に示す第一の実施例では、当然この操作は不要であるから、サインビット $\overline{x1}$ の入力を切り放して、「0」レベルに接続している。

ここで用いている全加算器4の動作は次式に示すとおりである。

$$S = A \oplus B \oplus Cin, Cout = A \cdot B + (A + B) \cdot Cin \quad \dots (3)$$

但し、 \oplus は排他的論理和、 \cdot は論理積、 $+$ は論理和を表わす。

第4図は本発明の第二の実施例を示す符号変換回路のブロック図である。

第4図に示すように、本実施例は μ LawPCM符号で構成された8ビットの入力信号($x1 \sim x8$)を反転する反転回路2と、この反転回路1の出力信号($\overline{x1} \sim \overline{x8}$)3の下位4ビット $\overline{x5} \sim \overline{x8}$ に論理「1」レベルの信号を上下1ビットづつ付加して生成した6ビットの信号5を入力し且つこの反転

サインビット16が「0」の時は、「-33」を、逆に、サインビット16が「1」の時は、「-34」を加算するようにしている。

〔発明の効果〕

以上説明したように、本発明の符号変換回路は、反転回路とビットシフト回路および加算器を用いることにより、従来のようにROMを使用することなく、純粋な組合せ論理回路のみで μ Law符号から綴形符号への符号変換が可能であるので、チップLSI内に組み込む際にも回路規模を大幅に圧縮し、チップサイズの小型化を実現できるとともにチップ価格を低減できるという効果がある。

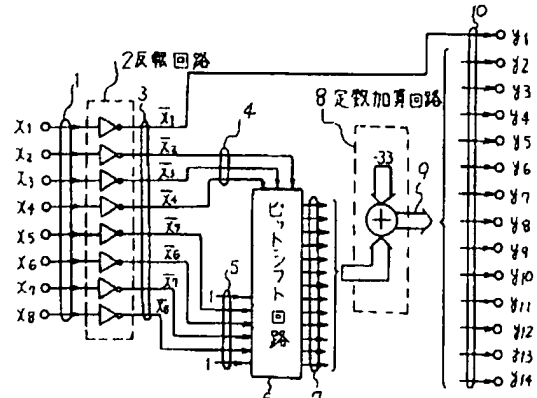
4. 図面の簡単な説明

第1図は本発明の第一の実施例を示す符号変換回路のブロック図、第2図は第1図に示すビットシフト回路図、第3図は第1図に示す定数加算回路図、第4図は本発明の第二の実施例を示す符号変換回路のブロック図、第5図は従来の一例を示

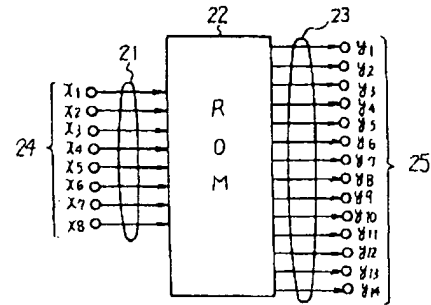
す符号変換回路のブロック図である。

1……8ビット μ Law符号入力信号、2……反転回路、3……反転回路出力信号、4……ビットシフト回路制御信号、5……ビットシフト回路入力信号、6……ビットシフト回路、7……ビットシフト回路出力信号、8……定数加算回路、9……定数加算回路出力信号、10……サイン+絶対値表示出力、11……1ビットシフト部、12……2ビットシフト部、13……4ビットシフト部、14……全加算器(FULL ADDER)、15……インバータ、16……サインビット、17……(1の)補数回路、18……2の補数表示出力。

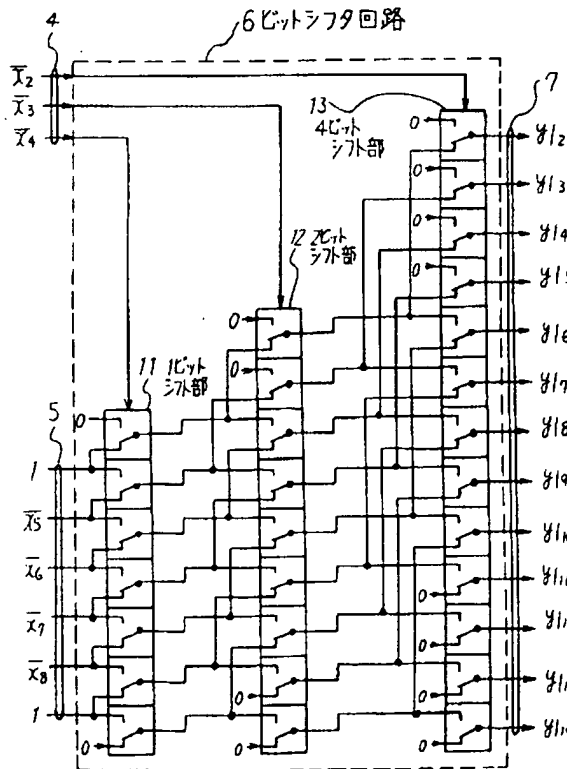
代理人 弁理士 内 原 晋



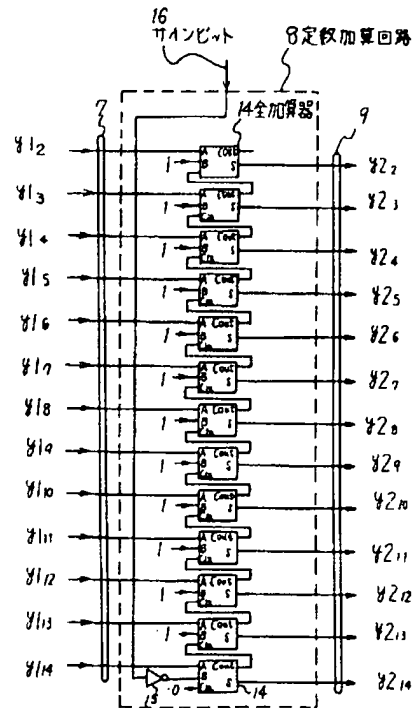
第 1 図



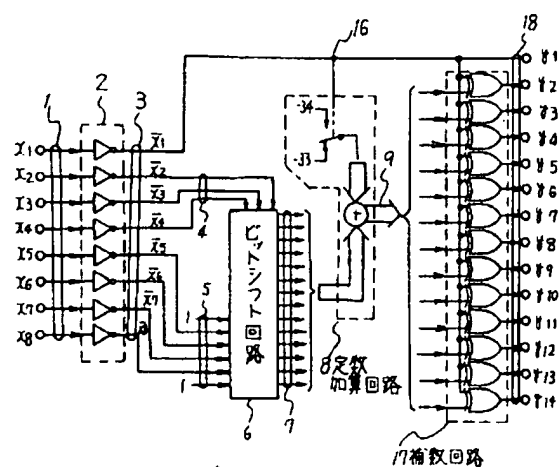
第 5 図



第 2 図



第 3 図



第 4 図